OOP

|  |  |
| --- | --- |
| Coupling & Cohesion | **Cohesion** refers to what the class (or module) can do. Low cohesion would mean that the class does a great variety of actions - it is broad, unfocused on what it should do. High cohesion means that the class is focused on what it should be doing, i.e. only methods relating to the intention of the class.  As for **coupling**, it refers to how related or dependent two classes/modules are toward each other. For low coupled classes, changing something major in one class should not affect the other. High coupling would make it difficult to change and maintain your code; since classes are closely knit together, making a change could require an entire system revamp.  Good software design has **high cohesion** and **low coupling**. |
| Name OOD principles that you know |  |
| Do you know SOLID principles (describe one of them) | 1. Single Responsibility Principle 2. Open Closed Principle 3. Liskov Substitution Principle 4. Interface Segregation Principle 5. Dependency Inversion Principle |